Fondamenti di Telecomunicazioni – Esercitazione 7

Corso di Ingegneria Fisica e Matematica
A.A. 2025–2026

Convenzioni.

- L'entropia di una sorgente discreta con alfabeto $\mathcal{A}$
$$H = -\sum_{x\in\mathcal{A}} p_x \log_2 p_x.$$
- La funzione entropia binaria è
$$h(p) = -p\log_2 p - (1-p)\log_2(1-p).$$
- La funzione $Q(\cdot)$ è
$$Q(\alpha) = \frac{1}{\sqrt{2\pi}}\int_{\alpha}^{+\infty} e^{-u^2/2}\,du.$$

Esercizio 1. Codifica di Huffman a blocchi e limite all'entropia

Sia data una sorgente discreta senza memoria con alfabeto
$$\mathcal{A} = \{x_1, x_2, \dots, x_M\}$$
e probabilità dei simboli
$$p_x, \qquad x \in \mathcal{A}.$$
L'entropia della sorgente, riferita al singolo simbolo, è
$$H = -\sum_{x\in\mathcal{A}} p_x \log_2 p_x.$$
Si considerino ora blocchi di lunghezza n, cioè parole del tipo
$$x_1 x_2 \cdots x_n \in \mathcal{A}^n.$$
Indichiamo con
$$p_{x_1 x_2 \cdots x_n}$$
la probabilità del blocco $x_1 x_2 \cdots x_n$, e con
$$H_n = -\sum_{x_1 x_2\cdots x_n \in \mathcal{A}^n} p_{x_1 x_2\cdots x_n} \log_2 p_{x_1 x_2\cdots x_n}$$
l'entropia della sorgente a blocchi di lunghezza n.

a) Mostrare che, essendo la sorgente senza memoria,
$$H_n = nH.$$

b) Sia $\bar{L}_n$ la lunghezza media di un codice di Huffman costruito sui blocchi di lunghezza n. Usare la disuguaglianza
$$H_n \le \bar{L}_n < H_n + 1$$
per mostrare che
$$H \le \frac{\bar{L}_n}{n} < H + \frac{1}{n}.$$

c) Concludere che

$$\lim_{n\to\infty} \frac{\bar{L}_n}{n} = H.$$

d) Definire l'efficienza della codifica a blocchi come

$$\eta_n = \frac{H_n}{\bar{L}_n}.$$

Mostrare che

$$0 \leq \eta_n \leq 1$$

e che

$$\lim_{n\to\infty} \eta_n = 1.$$

e) Commentare il significato operativo del risultato.

Svolgimento.

a) Poiché la sorgente è senza memoria, la probabilità di un blocco è il prodotto delle probabilità dei singoli simboli:

$$p_{x_1 x_2 \cdots x_n} = p_{x_1} p_{x_2} \cdots p_{x_n}.$$

Quindi

$$H_n = -\sum_{x_1 x_2 \cdots x_n \in \mathcal{A}^n} p_{x_1} p_{x_2} \cdots p_{x_n} \log_2 \left(p_{x_1} p_{x_2} \cdots p_{x_n}\right).$$

Usando la proprietà del logaritmo:

$$\log_2 \left(p_{x_1} p_{x_2} \cdots p_{x_n}\right) = \log_2 p_{x_1} + \log_2 p_{x_2} + \cdots + \log_2 p_{x_n}.$$

Allora

$$H_n = -\sum_{x_1 x_2 \cdots x_n \in \mathcal{A}^n} p_{x_1} p_{x_2} \cdots p_{x_n} \sum_{i=1}^{n} \log_2 p_{x_i}.$$

Separando i contributi:

$$H_n = -\sum_{i=1}^{n} \sum_{x_1 x_2 \cdots x_n \in \mathcal{A}^n} p_{x_1} p_{x_2} \cdots p_{x_n} \log_2 p_{x_i}.$$

Fissato un indice i, tutti i fattori con indice diverso da i si sommano a 1, perché

$$\sum_{x\in\mathcal{A}} p_x = 1.$$

Quindi ogni termine della somma fornisce

$$-\sum_{x_i\in\mathcal{A}} p_{x_i} \log_2 p_{x_i} = H.$$

Essendoci n termini uguali, si ottiene

$$\boxed{H_n = nH.}$$

b) Per un codice di Huffman costruito sui blocchi di lunghezza n, vale

$$H_n \leq \bar{L}_n < H_n + 1.$$

Usando il risultato precedente:

$$nH \leq \bar{L}_n < nH + 1.$$

Dividendo per n:

$$\boxed{H \leq \frac{\bar{L}_n}{n} < H + \frac{1}{n}.}$$

c) Poiché

$$\frac{1}{n} \to 0 \qquad \text{per } n \to \infty,$$

dal confronto

$$H \leq \frac{\bar{L}_n}{n} < H + \frac{1}{n}$$

segue

$$\boxed{\lim_{n\to\infty} \frac{\bar{L}_n}{n} = H.}$$

d) L'efficienza della codifica a blocchi è definita come

$$\eta_n = \frac{H_n}{\bar{L}_n}.$$

Poiché

$$\bar{L}_n \geq H_n = nH,$$

si ha

$$\eta_n \leq 1.$$

Inoltre $H \geq 0$, quindi

$$\eta_n \geq 0.$$

Dunque

$$\boxed{0 \leq \eta_n \leq 1.}$$

Infine, poiché

$$\frac{\bar{L}_n}{n} \to H,$$

si ottiene

$$\boxed{\lim_{n\to\infty} \eta_n = 1.}$$

e) Il risultato mostra che, codificando blocchi sempre più lunghi, la lunghezza media per simbolo

$$\frac{\bar{L}_n}{n}$$

si avvicina sempre di più all'entropia della sorgente H.

Il motivo è che la codifica di Huffman ha uno scarto massimo minore di 1 bit rispetto all'entropia, ma questo scarto è riferito all'intero blocco. Quando si divide per n, lo scarto massimo per simbolo diventa

$$\frac{1}{n},$$

che tende a zero.

In conclusione, la codifica a blocchi permette di avvicinarsi all'efficienza massima:

$$\eta_n \to 1.$$

Esercizio 2. Disuguaglianza di Kraft e sorgente con probabilità diadiche

Si consideri una sorgente discreta con alfabeto

$$\mathcal{A} = \{A, B, C, D, E\}$$

e probabilità

$$p_A = \frac{1}{2}, \qquad p_B = p_C = p_D = p_E = \frac{1}{8}.$$

a) Verificare che le probabilità sono diadiche, cioè della forma

$$p_i = 2^{-\ell_i}$$

per opportuni interi ℓ_i.

b) Determinare direttamente le lunghezze di codice

$$\ell_i = -\log_2 p_i.$$

c) Verificare che le lunghezze ottenute soddisfano la disuguaglianza di Kraft:

$$\sum_i 2^{-\ell_i} \leq 1.$$

d) Costruire un possibile codice prefisso con tali lunghezze e rappresentarlo tramite un albero binario.

e) Calcolare la lunghezza media del codice:

$$\bar{L} = \sum_i p_i \ell_i.$$

f) Calcolare l'entropia della sorgente:

$$H = -\sum_i p_i \log_2 p_i.$$

g) Commentare il risultato confrontando $\bar{L}$ e H.

Svolgimento.

a) Le probabilità sono tutte potenze negative di 2:

$$\frac{1}{2} = 2^{-1}, \qquad \frac{1}{8} = 2^{-3}.$$

Quindi la sorgente ha probabilità diadiche.

b) Le lunghezze naturali associate sono

$$\ell_i = -\log_2 p_i.$$

Dunque:

$$\ell_A = 1, \qquad \ell_B = \ell_C = \ell_D = \ell_E = 3.$$

c) Verifichiamo Kraft:

$$\sum_i 2^{-\ell_i} = 2^{-1} + 4 \cdot 2^{-3}.$$

Quindi

$$\sum_i 2^{-\ell_i} = \frac{1}{2} + 4 \cdot \frac{1}{8} = \frac{1}{2} + \frac{1}{2} = 1.$$

Pertanto

$$\boxed{\sum_i 2^{-\ell_i} = 1.}$$

La disuguaglianza di Kraft è soddisfatta con uguaglianza.

d) Poiché la disuguaglianza di Kraft è soddisfatta, esiste un codice prefisso con tali lunghezze.

Un possibile codice è:

Simbolo	p_i	ℓ_i	Codice
A	$\frac{1}{2}$	1	0
B	$\frac{1}{8}$	3	100
C	$\frac{1}{8}$	3	101
D	$\frac{1}{8}$	3	110
E	$\frac{1}{8}$	3	111

La costruzione si può leggere sull'albero binario seguente: a ogni ramo sinistro assegniamo 0, a ogni ramo destro assegniamo 1. Il codice di ciascun simbolo si ottiene leggendo le etichette dei rami dalla radice fino alla foglia corrispondente.

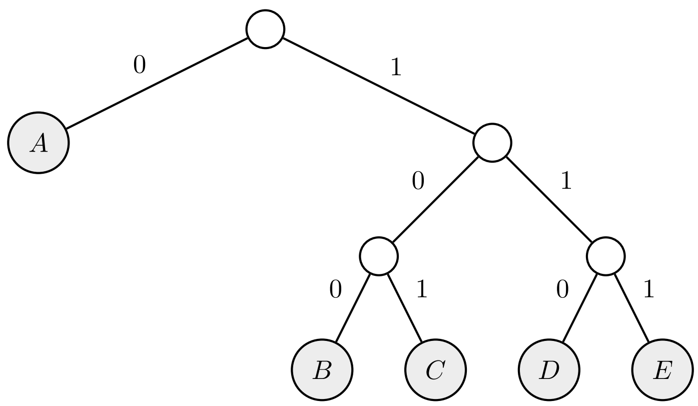

Figura 1: Albero binario associato al codice prefisso: $A \to 0$, $B \to 100$, $C \to 101$, $D \to 110$, $E \to 111$.

e) La lunghezza media vale

$$\bar{L} = \frac{1}{2} \cdot 1 + 4 \cdot \frac{1}{8} \cdot 3.$$

Quindi

$$\bar{L} = \frac{1}{2} + \frac{12}{8} = \frac{1}{2} + \frac{3}{2} = 2.$$

Dunque

$$\boxed{\bar{L} = 2 \text{ bit/simbolo.}}$$

f) L'entropia della sorgente è

$$H = -\sum_i p_i \log_2 p_i.$$

Quindi
$$H = -\frac{1}{2}\log_2\frac{1}{2} - 4\cdot\frac{1}{8}\log_2\frac{1}{8}.$$
Siccome
$$-\log_2\frac{1}{2} = 1, \qquad -\log_2\frac{1}{8} = 3,$$
si ottiene
$$H = \frac{1}{2}\cdot 1 + 4\cdot\frac{1}{8}\cdot 3 = 2.$$
Dunque
$$\boxed{H = 2 \text{ bit/simbolo.}}$$

g) In questo caso
$$\boxed{\bar{L} = H.}$$
Questo succede perché le probabilità sono diadiche: le lunghezze ideali
$$-\log_2 p_i$$
sono intere e quindi possono essere realizzate esattamente da un codice prefisso.

Di conseguenza il codice raggiunge il limite inferiore dato dall'entropia ed è ottimo.

Esercizio 3. Entropia binaria e codifica universale di sequenze binarie

Si consideri una sorgente binaria che genera blocchi di lunghezza n:
$$\mathbf{b} = (b_1, b_2, \ldots, b_n), \qquad b_i \in \{0, 1\}.$$
Si supponga che i bit siano i.i.d., ma che la probabilità dei bit 0 e 1 non siano note al codificatore.

Per codificare universalmente una sequenza $\mathbf{b}$, si procede così:

- si conta il numero k di bit uguali a 1;
- si codifica il valore di k;
- sapendo che la sequenza contiene esattamente k uni, si codifica quale tra i
$$\binom{n}{k}$$
pattern possibili è stato osservato.

a) Mostrare che il valore k può essere codificato usando circa
$$\log_2(n+1)$$
bit.

b) Mostrare che, una volta noto k, servono circa
$$\log_2\binom{n}{k}$$
bit per specificare il pattern osservato.

c) Usare l'approssimazione
$$\log_2\binom{n}{k} \simeq n\,h\!\left(\frac{k}{n}\right),$$
dove
$$h(p) = -p\log_2 p - (1-p)\log_2(1-p),$$
per interpretare il significato operativo dell'entropia binaria.

d) Spiegare perché codificare la posizione degli 1 oppure la posizione degli 0 richiede lo stesso numero di bit. Dedurre da questo la simmetria

$$h(p) = h(1-p).$$

e) Discutere per quali valori di k il numero di pattern possibili è massimo e per quali è minimo. Interpretare il risultato in termini della funzione $h(p)$.

f) Supponendo che la sorgente abbia probabilità del bit 1 uguale a q e spiegare perché, per n grande, con alta probabilità

$$\frac{k}{n} \simeq q$$

e quindi la lunghezza per simbolo della codifica è circa

$$h(q).$$

Svolgimento.

a) Il numero k di bit uguali a 1 può assumere i valori

$$k = 0, 1, 2, \ldots, n.$$

Quindi i valori possibili sono

$$n + 1.$$

Per descrivere uno tra $n+1$ valori servono

$$\lceil \log_2(n+1) \rceil$$

bit. Dunque, a meno dell'arrotondamento all'intero superiore, il costo è

$$\boxed{\log_2(n+1) \text{ bit.}}$$

Questo termine cresce lentamente rispetto a n, quindi per blocchi lunghi diventa trascurabile per simbolo:

$$\frac{\log_2(n+1)}{n} \to 0.$$

b) Una volta noto k, sappiamo che la sequenza contiene esattamente k uni e $n-k$ zeri.

Il numero di sequenze binarie di lunghezza n con esattamente k uni è

$$\binom{n}{k}.$$

Quindi, per individuare quale sequenza è uscita, servono circa

$$\boxed{\log_2 \binom{n}{k} \text{ bit.}}$$

La lunghezza complessiva della codifica è quindi circa

$$L(\mathbf{b}) \simeq \log_2(n+1) + \log_2 \binom{n}{k}.$$

c) Per n grande, usando l'approssimazione asintotica del coefficiente binomiale:

$$\log_2 \binom{n}{k} \simeq n\, h\left(\frac{k}{n}\right).$$

Quindi

$$L(\mathbf{b}) \simeq \log_2(n+1) + n\, h\left(\frac{k}{n}\right).$$

Dividendo per n:

$$\frac{L(\mathbf{b})}{n} \simeq \frac{\log_2(n+1)}{n} + h\left(\frac{k}{n}\right).$$

Per n grande,

$$\frac{\log_2(n+1)}{n} \simeq 0,$$

e quindi

$$\boxed{\frac{L(\mathbf{b})}{n} \simeq h\left(\frac{k}{n}\right).}$$

Questo dà un'interpretazione concreta dell'entropia binaria: $h(p)$ misura il numero di bit per simbolo necessari per descrivere una sequenza binaria in cui la frazione di uni è circa p.

d) Codificare la posizione degli 1 significa scegliere quali k posizioni, tra le n disponibili, sono occupate da uni:

$$\binom{n}{k}.$$

Codificare la posizione degli 0 significa invece scegliere quali $n-k$ posizioni sono occupate da zeri:

$$\binom{n}{n-k}.$$

Ma

$$\binom{n}{k} = \binom{n}{n-k}.$$

Quindi i due modi di codificare richiedono lo stesso numero di bit.

Usando l'approssimazione entropica:

$$\log_2 \binom{n}{k} \simeq n\, h\left(\frac{k}{n}\right),$$

e

$$\log_2 \binom{n}{n-k} \simeq n\, h\left(1 - \frac{k}{n}\right).$$

Poiché i due coefficienti binomiali sono uguali, si ottiene

$$\boxed{h(p) = h(1-p).}$$

Questa è la simmetria dell'entropia binaria.

e) Il numero di pattern possibili è $\binom{n}{k}$. Questo numero è minimo quando $k = 0$ oppure $k = n$. Infatti in questi casi c'è un solo pattern possibile:

$$\underbrace{00\cdots0}_{n \text{ zeri}}, \qquad \underbrace{11\cdots1}_{n \text{ uni}}.$$

Quindi

$$\binom{n}{0} = \binom{n}{n} = 1,$$

e servono $log_2 1 = 0$ bit per specificare il pattern.

Il numero di pattern è massimo quando

$$k \simeq \frac{n}{2}.$$

In questo caso ci sono moltissime sequenze possibili, perché il blocco contiene circa metà zeri e metà uni.

In termini di entropia binaria $h(p)$ è minima per $p = 0$ e $p = 1$, e massima per $p = \frac{1}{2}$. Infatti

$$h(0) = h(1) = 0, \qquad h\left(\frac{1}{2}\right) = 1.$$

f) Se la probabilità del simbolo 1 della sorgente è q, allora, per n grande, il numero di uni k sarà con alta probabilità vicino al suo valore medio:

$$k \simeq nq.$$

Quindi, con alta probabilità,

$$\frac{L(\mathbf{b})}{n} \simeq h\left(\frac{k}{n}\right) \simeq h(q).$$

Questo mostra che la codifica proposta è universale: anche senza conoscere a priori q, la lunghezza per simbolo si adatta alla frequenza empirica degli uni e tende al valore entropico corretto.

$$\boxed{\frac{L(\mathbf{b})}{n} \simeq h(q) \qquad \text{per } n \text{ grande.}}$$

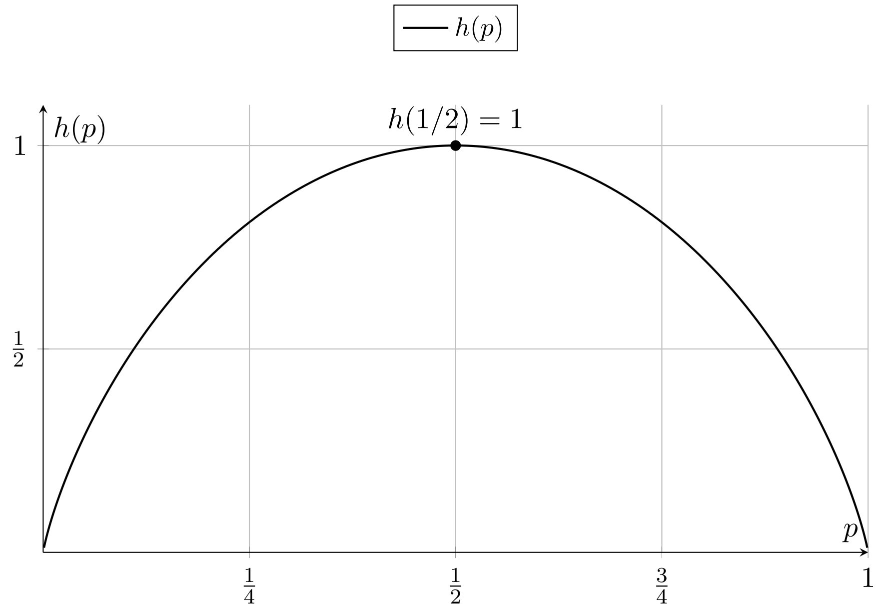

Figura 2: Entropia binaria: minima per sorgente deterministica, massima per $p = 1/2$.